ARI Research Note 94-19

Research Directions for Humans in Control of Automated Air Defense Command and Control Systems

John K. Hawley
Research Analysis and Maintenance, Inc.

Field Unit at Fort Bliss, Texas
John M. Lockhart, Acting Chief

Training Systems Research Division
Jack H. Hiller, Director

June 1994

United States Army
Research Institute for the Behavioral and Social Sciences

U.S. ARMY RESEARCH INSTITUTE FOR THE BEHAVIORAL AND SOCIAL SCIENCES

A Field Operating Agency Under the Jurisdiction of the Deputy Chief of Staff for Personnel

EDGAR M. JOHNSON
Director

Research accomplished under contract for the Department of the Army

Research Analysis and Maintenance, Inc.

Technical review by

David M. Johnson

Accesion For	
NTIS CRA&I	X
DTIC TAB	☐
Unannounced	☐
Justification	
By	
Distribution /	
Availability Codes	
Dist	Avail and / or Special
A-1	

NOTICES

DISTRIBUTION: This report has been cleared for release to the Defense Technical Information Center (DTIC) to comply with regulatory requirements. It has been given no primary distribution other than to DTIC and will be available only through DTIC or the National Technical Information Service (NTIS).

FINAL DISPOSITION: This report may be destroyed when it is no longer needed. Please do not return it to the U.S. Army Research Institute for the Behavioral and Social Sciences.

NOTE: The views, opinions, and findings in this report are those of the author(s) and should not be construed as an official Department of the Army position, policy, or decision, unless so designated by other authorized documents.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. AGENCY USE ONLY *(Leave blank)*

2. REPORT DATE
1994, June

3. REPORT TYPE AND DATES COVERED
Final Oct 92 - Sep 93

4. TITLE AND SUBTITLE
Research Directions for Humans in Control of Automated Air Defense Command and Control Systems

5. FUNDING NUMBERS
MDA903-92-C-0029
62785A
790
2222
C1

6. AUTHOR(S)
Hawley, John K.

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)
Research Analysis and Maintenance, Inc.
1790 Lee Trevino Drive, Suite 600
El Paso, TX 79936-4525

8. PERFORMING ORGANIZATION REPORT NUMBER
--

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
U.S. Army Research Institute for the Behavioral and Social Sciences
ATTN: PERI-IB
5001 Eisenhower Avenue
Alexandria, VA 22333-5600

10. SPONSORING/MONITORING AGENCY REPORT NUMBER
ARI Research Note 94-19

11. SUPPLEMENTARY NOTES
Contracting Officer's Representative, John M. Lockhart.

12a. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release;
distribution is unlimited.

12b. DISTRIBUTION CODE
--

13. ABSTRACT *(Maximum 200 words)*

This report describes work performed during the second year of the project titled "Command and Control Decision Making Requirements During Engagement Operations." The work involves the development of a human performance and training testbed for automated air defense command and control. The testbed is a flexible simulation capability for the study of issues involving human performance and training in a complex supervisory control setting. The first portion of the report addresses the testbed's objectives and integration concept. Next, the report outlines a concept for human supervisory control of a complex, automated process control environment. This concept is referred to as intelligent rule-based supervisory control, or IRBSC. IRBSC involves cooperative control of a real-time process by human operators and an expert system embedded in the command and control computer. Finally, the report outlines a research agenda for using the testbed to explore human performance, training, and performance support issues for real-time command and control systems.

14. SUBJECT TERMS

Air defense	Supervisory control
Command and control	Rule-based supervisory control
Automation	Training human performance

15. NUMBER OF PAGES
27

16. PRICE CODE
--

17. SECURITY CLASSIFICATION OF REPORT	18. SECURITY CLASSIFICATION OF THIS PAGE	19. SECURITY CLASSIFICATION OF ABSTRACT	20. LIMITATION OF ABSTRACT
Unclassified	Unclassified	Unclassified	Unlimited

NSN 7540-01-280-5500

Standard Form 298 (Rev. 2-89)
Prescribed by ANSI Std. Z39-18
298-102

RESEARCH DIRECTIONS FOR HUMANS IN CONTROL OF AUTOMATED AIR DEFENSE COMMAND AND CONTROL SYSTEMS

CONTENTS

	Page
INTRODUCTION	1
THE PROBLEM	1
THE APAWS TESTBED	5
Concept	5
Program Status	6
APAWS RESEARCH AGENDA	8
Human Performance	8
Training and Performance Support	15
DISCUSSION	17
REFERENCES	21

LIST OF FIGURES

Figure	1.	APAWS integration concept	6
	2.	A taxonomy of supervisory control performances	10
	3.	RBSC operational cycle	12

RESEARCH DIRECTIONS FOR HUMANS IN CONTROL OF AUTOMATED AIR DEFENSE COMMAND AND CONTROL SYSTEMS

1.0 OVERVIEW

This report describes work performed during the second year of Contract Number MDA903-92-C-0029, "Command and Control Decision Making Requirements During Engagement Operations." The work described in the report involves the development of a human performance and training testbed for automated air defense command and control. In the present usage, the term testbed refers to a flexible simulation capability that can be used to study a range of issues involving human performance and training in a complex supervisory control setting. The first portion of the report addresses the testbed's objectives and integration concept. Next, the report outlines a concept for human supervisory control of a complex, automated process control environment. This concept is referred to as intelligent rule-based supervisory control, or IRBSC. IRBSC involves cooperative control of a real-time process by human operators and an expert system embedded in the command and control computer. Finally, the report outlines a research agenda for using the testbed to explore human performance, training, and performance support issues for real-time command and control systems.

2.0 THE PROBLEM

Information technology is precipitating a revolution in warfighting doctrine and tactics and in weapon systems themselves. The future warfighting environment will be geographically and temporally dispersed (non-contiguous in time and space in the terminology of the new FM 100-5) and populated by numerous small, mobile, and semi-autonomous units possessing weapons of incredible accuracy and lethality. We have been provided with a peek at this future warfighting environment during the recent Gulf War. Desert Storm may have been brief, but it likely has changed the face of warfare in the same way that the German blitzkrieg offensives into Poland, France, and Russia changed the face of warfare some 50 years ago. We have come from blitzkrieg to AirLand Battle (the son of blitzkrieg) to AirLand Operations (the grandson of blitzkrieg and the focus of FM 100-5) with a 50-year time span. Moreover, the wheel is still turning, ever faster, driven by advances in computing, sensor, communications, and the other components of information technology.

The new information-technology-based weapons recently employed during the Gulf War are complex sociotechnical systems that include both human and machine components. Recent developments in information technology have paced a rapid evolution on the machine (i.e., hardware and software) side of weapons system operations. As machine technology has evolved, the operator's role in many of these systems also has changed. Previous systems require operators to perform in a tradition manual control role. That is, humans has primary

responsibility for perception, decision making, response selection, and response execution. In contemporary systems, the human operator's role is vastly different. Instead of direct participation in the control process, the operator's role increasingly is one of monitoring a computer controller and intervening in the case of abnormal situations. Put another way, the operator's role has shifted from traditional operator to supervisory controller, sometimes referred to as a system manager. It is easy to spot the evidence of this role change in weapons like the Patriot air defense missile system or the Comanche attack helicopter and in various command and control systems. The change is, however, also apparent in less technically sophisticated weapons systems and in the command and control support provided to small unit leaders.

Nowhere is the trend toward the widespread use of information technology and its offspring automation more apparent than in command and control for High and Medium Altitude Air Defense (HIMAD) systems. By HIMAD, we are referring to the present Hawk, AN/TSQ-73 (Q-73) Missile Minder, and Patriot systems as well as the emerging Air Defense Tactical Operations Center (ADTOC), National Missile Defense (NMD), Theater High Altitude Air Defense (THAAD), and Corps Surface-to-Air Missile (Corps SAM) systems. The marked increase in weapons lethality and threat approach speeds faced by these systems (e.g., in the form of Tactical Ballistic Missiles — TBMs) requires that the engagement process be augmented by technology. Operators must have computer-based support to rapidly and simply provide the information necessary for engagement decision making. The time windows involved in present and future command and control operations are simply too short to consider any other approach. Also, real-time interaction between the human operators and the computer to exercise system control is essential to effective employment of this class of systems (US Army Air Defense Artillery School, 1991; 1992). The term real-time, in the present context, refers to a situation in which the system responds immediately at the time an event occurs. Real-time systems are characterized by rapid and frequent interactions between the operators and the computer controller.

Although automation is viewed as essential to future air defense command and control operations, the impact of automation on human operators has not always been positive. There is increasing evidence that poorly human engineered automated systems suffer from a number of problems that can result in decreased system effectiveness or even catastrophic system failure. The human performance problems associated with what is often termed "clumsy" automation generally fall into one of two categories: (1) loss of situational awareness and (2) skill decay. The essential idea of situational awareness is that operators must keep track of a lot of information from a variety of sources over time and organize and interpret this information to behave appropriately (Howell, 1993). As tasks are given to an automatic controller, the operators' interaction with the system is reduced. Consequently, when an abnormal situation does occur and requires operator action, the operator may be slow to detect it and may take too long to decide upon the appropriate control actions. Contrary to much current thinking, the requirement for operators to maintain situational awareness is not eliminated in an automated

system. In fact, some automation styles can make it more difficult for them to maintain awareness. The clumsy use of automation to eliminate human error can become the source of new types of error or system failure. Moreover, the cost of clumsy automation often becomes most apparent during critical events and under high operating tempo conditions.

There also appear to be longer-term consequences of being removed from the control loop. As they receive less and less hands-on experience, operators can lose proficiency in basic control operations. When called upon to intervene, their skills may have decayed to the point where they cannot execute the proper control sequence in a timely manner. An increasing body of research and experience indicate that effective supervisory control requires a skilled operator in somewhat continuous and meaningful interaction with the controlled process. The problem is that we do not know how to bring about continuous and meaningful interaction with the controlled process without eliminating some of the positive aspects of automation or introducing new types of errors or failure modes.

Recent advances in information technology have resulted in several potential solutions to problems of human performance in automated systems. The first of these is flexible automation. Under a flexible automation regimen, both the level and style of automation are variable as a function of operating conditions. Initially, operators can determine their desired automation mode on-line and often can select from several options. Later on, they can change the automation scheme in real-time as the situation requires.

Display format adaptivity is the second technological advance having significant potential for improved person-machine integration in automated systems. Adaptive displays are variable in format or logic as a function of mission stage or operating conditions. Although there is no absolute requirement that they be used together, flexible automation often involves the use of displays attuned to the automation mode and operating conditions (Shanit, Chang, and Salvendy, 1987). Hence, a reference to flexible automation often implies that control station displays are adaptive.

Prior to proceeding with the present discussion, it is necessary to clarify some terms that are used throughout the report. As noted above, the term flexible automation means that the level and style of automation are variable as a function of the mission stage or operating conditions. Flexible automation is achieved through the use of dynamic function allocation (and re-allocation) along with adaptive control station displays. Dynamic function allocation means that the boundary defining the person-machine interface is not fixed. Rather, the boundary between operator and computer can be changed in real-time to accommodate operating requirements. A flexible automation regimen that permits the operator to hand off tasks to the computer during periods of high load with the option to later take them back is often referred to as a task-offloading aid (Kirlik, 1993). A cruise control mechanism in an automobile or an autopilot system in an aircraft are common examples of task-offloading aids.

Flexible automation technology creates the possibility for a "personalized" soldier-machine interface tailored to mission requirements and to individual operator preferences. Dynamic allocation and adaptive displays represent potential solutions to the problems of situational awareness and skill decay. However, recent research suggests that the introduction of a task-offloading aid eliminates some task demands but also creates new ones. At a minimum, for example, the operator must program, engage, and disengage the aid (Kirlik, 1993). Wood (1993) also comments that windows-oriented adaptive displays tend to provide operators with multiple "keyhole" views of the world while denying them any "peripheral" vision. He further remarks that operators often require a comprehensive "navigation" system to maintain situational awareness in systems employing extensive windowing capabilities.

The crux of the previous discussion is that the introduction of automation in air defense command and control and other process control applications has not been without problems. Information technology offers various solutions to the difficulties that we have encountered thus far in our excursion into the unfolding world of human-automation interaction. New technologies introduced to eliminate the difficulties associated with the use of older technologies often bring with them obstacles of their own — as illustrated above. In the air defense command and control arena, we are concerned with operator performance in a dynamic, very complex setting. The fundamental premise of engineering psychology is that one cannot study humans in isolation from the tools they use. Further, in the case of air defense command and control, operators do not do things alone — they function as part of a team. To compound this latter problem, we also now have a situation in which one of the team members is a computer.

Because of the circumstances noted above and because we know so little about the effects of many of these potential system features on operators' cognitive processes and thus on system performance, empirical research must be a critical aspect of the concept, materiel, and training development processes for future systems. At present, however, there are few facilities suitable for conducting the kinds of research needed to address the issues of the proper role for and training of operators in automated air defense command and control systems. Such a facility must permit: (1) rapid incorporation of advanced design features such as dynamic function allocation and adaptive displays; (2) low-cost application of contemporary performance support technologies such as knowledge-based processing and neural nets; and (3) flexible, team-oriented operations. Above all, the capability must be low-cost. Current military budgets will not support the development of costly research and development facilities. In the next section, we describe a supervisory control research facility designed specifically with these requirements in mind.

3.0 THE APAWS TESTBED

3.1 Concept

In response to the problems cited in the previous section, Research Analysis and Maintenance, Inc. (RAM), under contract to the US Army Research Institute (ARI), has initiated a multi-year research effort concerned with human performance, training, and performance support in automated air defense command and control operations. The focus of this program is the impact of automation on air defense command and control operators and the consequences of their role change from traditional operators to supervisory controllers. To investigate these issues as they relate to future air defense command and control systems, the first portion of the effort concerns the development of a human supervisory control performance and training testbed — denoted APAWS, for ARI PC-Based Analytical Workstation — tailored for air defense command and control applications.

The developmental concept for APAWS is illustrated in Figure 1. Key elements of the platform's design concept include:

- Software integration versus software development,
- Re-use of proven software modules,
- Use of the Ada programming language,
- Hosting the system on a PC-class platform (80486/50), and
- Open, hardware-independent software architecture.

The APAWS developmental strategy is intended to reduce developmental risk, time, and cost.

When completed, the basic APAWS capability will provide air defense decision makers with a platform capable of emulating potential concepts of operations for both current and future systems. The finished platform will support: (1) dynamic soldier-machine function allocation, (2) adaptive and reconfigurable control station displays, and (3) an embedded performance assessment capability (PAC). The APAWS PAC will be developed following an approach to Patriot operator performance assessment proposed in Hawley, Howard, and Martellaro (1982) and later refined and implemented by Brett and Allender (1990). All user documentation and on-line performance support will be available in a hypertext format. APAWS will be a "paperless" research environment.

In addition to the basic capabilities shown in Figure 1, the APAWS platform is designed with a number of growth capabilities in mind. From the perspective of supervisory control research and development, several of the most significant of APAWS' pre-planned growth paths include: (1) the ability to include a multi-node command and control configuration through the use of local area network (LAN) technology, (2) the use of speech synthesis technology, and (3) simulated participating units (air targets, other command and control nodes, etc.) based on an

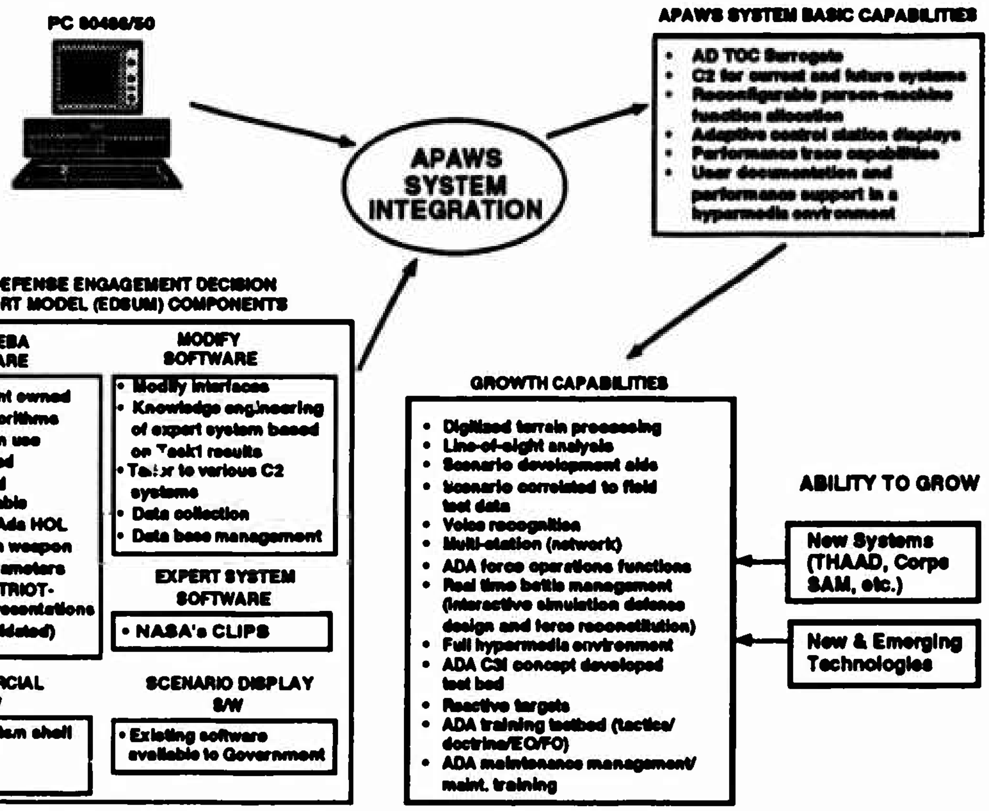

Figure 1. APAWS integration concept.

embedded expert system. The expert system chosen for implementation in APAWS is the Ada version of NASA's CLIPS (C-Language Integrated Production System). The embedded expert system also will serve as the basis for (1) flexible automation through dynamic function allocation, (2) intelligent performance support features (i.e., job performance aids — JPAs), and (3) embedded training. The APAWS platform with CLIPS embedded will provide air defense decision makers with a vehicle for exploring concepts of operation for an explicitly rule-based command and control system.

3.2 Program Status

The APAWS platform is under development in three progressive stages referred to as prototypes. Prototype I was completed in January 1993. Stage one consisted of: (1) the Ada-based TEWA (Threat Evaluation and Weapons Assignment) model operating in real time and (2) a reconfigurable graphic user interface (GUI). The TEWA model is an Ada-based version of the command and control logic embedded in the Q-73 system. The Q-73 is capable of providing command and control for Hawk, Patriot, and composite Hawk-Patriot air defense missile battalions. TEWA was developed initially as a batch-run concept and system evaluation tool. In its present real-time form, TEWA constitutes an interactive command and control system simulation model. The TEWA model thus provides a functional command and control baseline for the APAWS testbed. Since it is written in Ada, TEWA readily can be modified to represent the logic of other

command and control systems or concepts.

The second APAWS prototype is scheduled for completion at the end of the current contract year (September 1993). At this stage, the TEWA model will be integrated with the adaptive control station display to form a reconfigurable air defense command and control tactical operations simulator (AD C2 TOS). Prototype II will also support a run-time-adaptive operator PAC. A run-time-adaptive PAC is one in which users can determine at run time — by selecting from a menu of options — the operator Measures of Performance and soldier-machine Measures of Effectiveness to be recorded.

Developing APAWS Prototype III will involve integrating the CLIPS expert system into the generic AD C2 TOS. Rules governing a portion of the Engagement Operations function set for command and control of a Patriot- or Corps-SAM-like air defense missile system will also be developed and exercised as a test case. APAWS Prototype III will provide a platform for conducting empirical research on concepts of operation for humans in control of future air defense systems. The platform will also provide a vehicle for examining the impact of various supporting technologies applied to air defense command and control. Examples of potential technologies that could be used in this respect include neural networks (e.g., as a non-cooperative target recognition [NCTR] aid), hypermedia (e.g., text, graphics, animation, and sound), and fuzzy-logic-based rule processing. Prototype III was scheduled for completion by April 1994. However, contract was terminated due to lack of funds.

In addition to the features described above, the embedded expert system used in Prototype III can also be used to augment the baseline TEWA model. Portions of any prospective command and control logic not presently represented in TEWA can be developed using the CLIPS portion of the APAWS software, as opposed to being "hard coded" into the TEWA model using the Ada programming language. The ability to enhance APAWS using CLIPS as opposed to Ada software modules will increase the testbed's flexibility as a research tool and significantly reduce the software development time required for system enhancements.

One of the problems often encountered when conducting research on new or hypothetical systems is finding or developing a suitable test operator population. The command and control concepts likely to be evaluated using APAWS will not exist at the time test runs are conducted. Consequently, there will be no well-trained operator population from which to select test subjects. By the same token, developing our own cadre of test operators would be a lengthy process. In the APAWS effort, we intend to circumvent this problem by developing test scenarios that, in essence, represent a sophisticated air defense command and control game. The term game, in the present context, refers to a simulation that preserves the functionality of the target environment but omits the detail and specificity of the real-world situation. It is the detail and specificity of the real-world performance environment that results in lengthy training times. In this manner, test subject performance should stabilize rather quickly (e.g., two to three days), and

experimental results should still generalize to the real-world performance situation. We followed a similar strategy in an earlier research effort involving Patriot operators and met with encouraging results (see Hawley *et al.*, 1982).

4.0 APAWS RESEARCH AGENDA

As noted in the previous section, APAWS is intended as a human performance and training testbed tailored for air defense command and control applications. Had it been completed, the testbed would have been evaluated in a series of verification and validation (V&V) exercises. After V&V testing, the first round of soldier-automation experiments using the testbed would have begun. Our research agenda for automation and supervisory control in air defense command and control operations would have subsumed two topic areas: (1) human performance, and (2) training and performance support. Although these topics are related, each area is addressed separately in the sub-sections to follow.

4.1 Human Performance

When considering human performance requirements in supervisory control, it is instructive to begin with Rasmussen's (1986) supervisory control taxonomy. Under Rasmussen's taxonomy, human tasks in a process control setting can be classified into one of three categories — skill-based behavior (SBB), rule-based behavior (RBB), and knowledge-based behavior (KBB). SBB consists of sensory and motor performances during acts that, after a statement of intent, take place without conscious control as smooth, automated, and highly integrated behaviors. An example of SBB is entering instructions into a command and control computer.

In RBB, the task sequence is *consciously* controlled by a stored rule. This governing rule may have been (1) derived empirically during previous operations, (2) communicated from another person's know-how, or (3) prepared on occasion through conscious problem solving and planning. The boundary between SBB and RBB is not distinct. It depends on both the level of training and attention of the operator. Hence, RBB for an inexperienced operator might be SBB for a more experienced one. Also, a task that begins as RBB and through practice transitions to SBB is said to have been trained or performed to automaticity.

When operators are faced with a situation for which no explicit rules are available, behavioral control moves to a higher conceptual level in which performances are goal-oriented and structured on occasion through conscious problem solving and planning. Rasmussen refers to this latter category of human performance as KBB. Mission planning, complex problem-solving, and trouble-shooting are common examples of KBB.

Rasmussen's taxonomy provides a useful perspective on the human performance requirements underlying supervisory versus traditional control. Simply stated, a supervisory control regimen emphasizes and retains operator decision-

making and problem-solving tasks (i.e., KBB and upper-level RBB) while relegating most direct sensory and psychomotor tasks and many rule-based performances (i.e., lower-level RBB and SBB) to machine subsystems. By definition, the knowledge-based performance domain remains the exclusive preserve of the human operators. At the other end of the performance spectrum, activities in the skill-based domain can be allocated either to humans or to the machine. Any skill-based performances assigned to human operators should be structured to be as error-proof as possible.

Sheridan (1992) argues that the essence of supervisory control is partitioning control intelligence between the human and machine components. In line with the discussion in the previous paragraphs, the problem of allocating the so-called intelligent aspects of system control between humans and the machine reduces to one of treating RBB and KBB. Handling simple RBB, the lower level of Rasmussen's rule-based performance domain, is not particularly problematic. Simple rules that do not change or that are applied universally across objects can be hard-coded into machine software. The real problem in partitioning control intelligence involves handling what might be termed meta rules — the top level of the rule-based performance domain and the lower-level of the knowledge-based domain.

Meta rules are second-level rules describing how to use lower-level rules. In situations like air defense command and control where much of the process can be described in terms of blocks of simple rules that are directed at specific ends (e.g., track identification, track prioritization, track engagement, etc.), meta rules often represent a set of higher-order rules describing when to execute a specific lower-level rule block. When expert system users complain that a particular decision aid is "trivial" or "not robust," what they really are saying is that the rule base consists only of simple rules. By itself, the expert system is not able to resolve many complex decision situations. It is this feature of applied expert systems that often requires that they be developed in layers of increasing complexity (see Obermayer, 1991).

As with SBB and RBB, we have observed that the dividing line between RBB and KBB is not distinct; one type grades into the other. Both upper-level RBB and lower-level KBB involve meta rule processing. The primary difference is one of rule complexity. Meta rule processing gets fuzzier and more involved as additional knowledge-based elements come into play. Finally, rule construction and handling gets so involved that it exceeds the capacity of current knowledge-based processors. At this point, we are in the domain of strict KBB, and a human operator must assume responsibility for the performances in question. The relationships among the various aspects of RBB and KBB are illustrated in Figure 2.

Following the logic of the previous paragraphs, one way of viewing flexible automation (defined in terms of dynamic function allocation) is in terms of partitioning and re-partitioning the shaded portion of the performance set shown in Figure 2 between human operators and the machine subsystem. The notion of defining supervisory control in terms of partitioning rule-based performances between humans and the machine is not new. A control regimen in which rule-

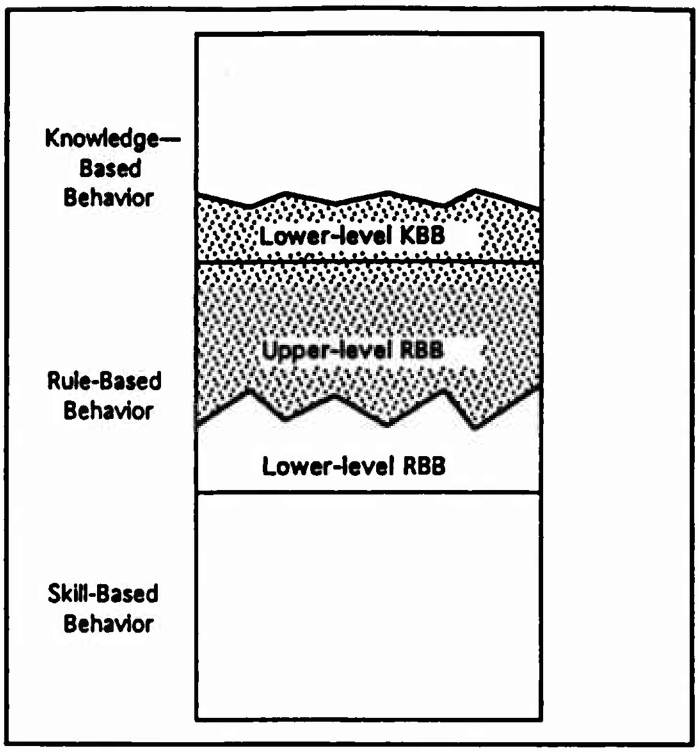

Figure 2. A taxonomy of supervisory control performances.

based performances are explicitly partitioned between human operators and the machine is formally referred to as rule-based supervisory control, or RBSC (see Hamill and Gersh, 1991; Gersh and Hamill, 1991; Hamill and Gersh, 1992). In an RBSC system, the operator/decision maker issues commands to the system in the form of condition-action (i.e., IF...THEN) production rules.

Many contemporary control systems employ rule-based processing. However, in most such systems, the rule base and processing steps are more or less invisible to users. In the air defense command and control arena, for example, the Q-73 and Patriot systems both employ rule-based processing but much of that logic is hard-coded in software and not apparent to operators. The aspect of RBSC that makes it different from the processing on the Q-73 or Patriot is that the decision maker explicitly formulates supervisory control commands in the form of condition-action rules and then monitors and adjusts the system as it applies those rules to the control situation.

Hamill and Gersh's original formulation of the RBSC paradigm required operators to formulate and re-formulate the conditions and actions comprising the control rule set. Requiring operators to explicitly formulate, re-formulate, and directly input the control rule set in real time may not, however, be the most effective operational mode for RBSC. In our view, a preferred approach to implementing RBSC is to permit the operators to partition a set of rules defining the command and control universe. A subset of this rule set is assumed by the operators and the complement is assigned to the machine. The subset assigned to

the machine is tailored to the local situation by setting parameters and conditions during system initialization. Similarly, replanning involves modifying the composition of the rule subsets or adjusting the parameters and conditions of the subset assigned to the machine. New rules are developed and added to the command and control superset in response to systematic inadequacies in the system's performance.

Implementing the variant of RBSC described in the previous paragraphs will first require creating a dynamic repository for the command and control rule superset. Human operators will interact with this repository in real time to establish and adapt the system control strategy. At present, the obvious choice of a repository for the command and control rule superset is an expert system embedded in the computer controller. We refer to an RBSC regimen implemented through an embedded expert system as Intelligent RBSC, or IRBSC (see Hawley, Strub, and Lockhart, 1993). Rule-based processing using an embedded expert system also enables the explicit handling of meta rules. Hence, complex rule-based performances that might strictly be classified as KBB increasingly can be assigned to the machine for handling by an embedded expert system or decision models based on other forms of artificial intelligence such as a case-based reasoning tool (Swamidass, 1993).

To further illustrate the RBSC concept, consider how Patriot would function as an RBSC system. To begin, operators would select a level of automation by partitioning the command and control rule base. The level of automation could be set anywhere between the present semi-automatic and automatic modes. Displays also could be tailored to the expected tactical situation or operator preferences. Tactical initialization would be explicitly framed in terms of setting parameters and conditions for rules assigned to the weapons control computer (WCC). Once an air battle began, real-time performance indices would be monitored to gauge the effectiveness of the defensive strategy. If the defensive strategy does not produce the desired results, an adjusted strategy would be formulated, evaluated, and implemented in near-real-time through (1) a change in the level of automation or (2) adjustments to the parameters and conditions of the rules assigned to the WCC.

As multiple air battles are completed, defense planners at the ADTOC would note systematic inadequacies in the system's performance against various classes of air threats. During lull periods, new or modified rules to compensate for these inadequacies would be developed, evaluated, and entered into the command and control rule superset using an embedded expert system. Command and control software development and modification as currently performed would not be required. Large portions of software-based firing doctrine would be coded as production rules instead of traditional computer code. Further, instead of a linear, sequential series of operations, air battle planning and management would consist of two loops. The first loop would be a Monitor-Intervene cycle for short-term, tactical adjustments; and the second loop would be a Learn-Teach cycle for longer-term, strategic changes to firing doctrine, tactics, techniques, and procedures (See Sheridan [1992] for a discussion of the Plan-Monitor-Intervene-Learn-Teach cycle

of human supervisory control actions.) Figure 3 illustrates this RBSC operational cycle.

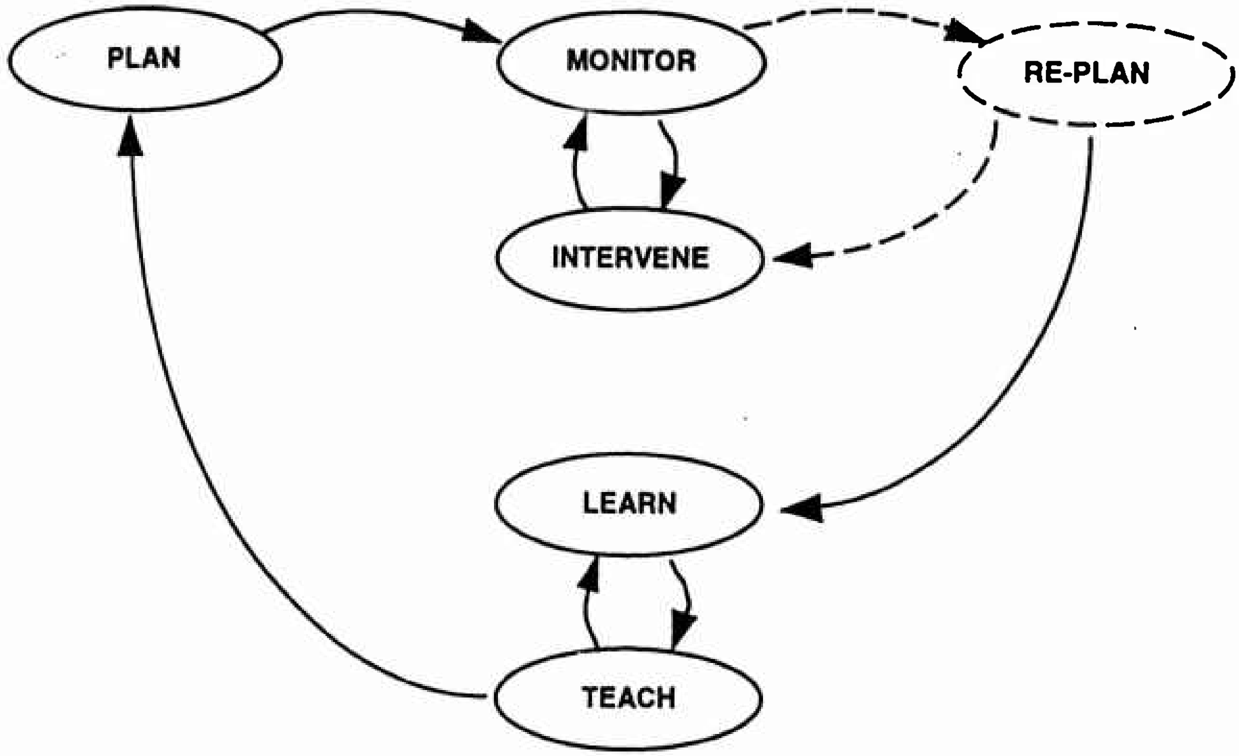

Figure 3. RBSC operational cycle.

Sheridan (1992) remarks that the human interactive computer in a supervisory control setting has two primary functions: (1) command and control and (2) decision aiding. In IRBSC, the embedded expert system participates in both of these activities. APAWS's embedded expert system supports command and control by functioning as a task-offloading aid; it facilitates operator decision aiding by serving as an information source and intelligent JPA.

As noted above, APAWS Prototype III will contain an embedded expert system. This embedded expert system will provide the basis for dynamic function allocation through dynamic partitioning of the control rule base between human players and the APAWS computer. Under normal operating conditions, the expert system will be structured to handle all aspects of system control without human intervention. During tactical initialization, the operators will determine a function split (i.e., a level of automation) by leaving one subset of functions assigned to the expert system and assuming responsibility for the complementary subset. The operators are responsible for tailoring the overall control strategy by setting conditions and parameters for the subset assigned to the machine. They will also establish the automation style by defining the protocol for communicating with the expert system. The human-expert system protocol specifies the conditions under which the expert system will interact with the operators and the format of these communications.

As the situation unfolds, the operators will adapt their control strategy by (1) resetting rule parameters and (2) task off- or on-loading (i.e., changing the composition of the rule subset assigned to the machine). Operators will also be able to modify the automation style by changing the protocol for operator-expert system interaction. APAWS will permit operators to select automation modes from a set of pre-specified options or they can tailor the person-machine interface by defining their own individual styles. For example, if the operators desire to assume a more direct role in system operations, they will accept responsibility for processing initially assigned to the machine. Similarly, under conditions of heavy loading, they may off-load some of their control responsibilities to the embedded expert system through trading, sharing, or cooperative control (Sheridan, 1992). Shared control means that the operators and the computer control different aspects of the system at the same time. Under a shared control regimen, the computer is used to extend the operators' capabilities. Trading control refers to a situation where the computer backs up or completely replaces the operators. Backing up the operators means that the computer picks up the slack for the operators when they falter. In a cooperative mode, system control is initiated by one party (operators or the computer) and the other then refines it.

APAWS's flexible automation capability will be available universally across control functions and entities or on a function group by function group or entity group by entity group basis. That is, operators will have the flexibility to apply a single automation mode to the entire command and control rule set or to "unbundle" the function set and establish different automation modes for major clusters of control functions. Similarly, operators can apply a single automation mode to all tracks or they can unbundle the track set and apply different modes across track subsets. For example, under selected conditions, operators might choose to assume a near-manual role in the Track Identification (ID) process for some or all of the tracks while permitting the machine to handle Track Engagement tasks automatically once a track's ID has been established.

We noted earlier that APAWS is intended as a human performance and training testbed adapted for an air defense command and control setting. The previous paragraphs describe our developmental concept for APAWS Prototype III. We have a general idea of how we want the testbed to perform, but there are a number of developmental issues remaining to be resolved before Prototype III is ready for use. Several of the most significant of these developmental issues are discussed in the paragraphs to follow.

A taxonomy of flexible automation. As discussed above, one of the capabilities planned for APAWS is flexible automation — defined in terms of dynamic function allocation and adaptive control station displays. If we are to explore the impact of flexible automation on human and system performance, we must know how to define and manipulate it as an experimental variable. In this respect, we allude to several of the dimensions defining flexible automation, namely Level and Style. Level is defined in terms of the degree of control exercised by the machine. Style refers to the manner in which the expert system and the human

operators interact. Another dimension that we are exploring is termed Universality. Universality refers to whether a single automation mode is applied across control functions and entities or whether the control function and entity sets are unbundled with various clusters of functions or entities being assigned different automation modes. Our early attempts to define flexible automation modes in terms of these three dimensions suggest that other factors are also involved. One of our research objectives is thus to continue refining a taxonomy characterizing flexible automation.

Operator-automation integration. Effective joint process control by human operators and an expert system will require that they work together in a smooth or "seamless" fashion, much like a well rehearsed human team. Research indicates that team members in high-performance, all-human teams adjust their working styles to compensate for each other's strengths and weaknesses. The degree to which this level of cooperation can be achieved when one team member is an intelligent machine is an open question. The issue of human-computer *cooperation* (as opposed to simple interface) in process control will become more important as expert systems and other types of knowledge-based processing are increasingly used in system control. Wood (1993) argues that research in this area must move beyond a simple concern for human-computer interface into the area of human-computer cooperation.

A new look at operator performance requirements in automated systems. Sheridan (1992) and others have identified and described generic residual operator performance requirements in automated processing (i.e., Monitor, Intervene, Learn, Teach, and Plan). These performance requirements were described at a time when automation involved a fixed allocation of functions between humans and the machine. In our brief experience with APAWS, flexible automation and the use of explicit knowledge-based processing support imply human performance requirements beyond those described by Sheridan. His general categories of residual operator activities still are valid but the actions involved in the performance of each are somewhat different. Kirlik (1993) notes, for example, that a task off-loading aid requires operators to develop and implement a *strategy* for selecting the mode of control based on an assessment of task demands and performance requirements. His research indicates that the strategy the operator develops for managing interaction with the task off-loading aid is the most significant factor in (1) the use or non-use of such an aid and (2) its impact on system performance. Current treatments of human performance requirements in an automation setting (including Sheridan's) do not address the topic of strategy development.

Our human performance research agenda is directed squarely at alleviating the problems of loss of situational awareness and skill decay that have traditionally accompanied automation. We are not sure that flexible automation and associated concepts such as RBSC will be any more successful in combating these human performance problems than previous technological interventions. Flexible automation concepts are, however, rapidly being introduced into the industrial

automation arena, and it is only a matter of time before they are proposed for use in real-time military command and control. With APAWS, we are seeking to develop a research facility in which these concepts and their associated training strategies can be rapidly and inexpensively tested and debugged before committing to full-scale development.

4.2 Training and Performance Support

Our second category of research objectives concerns training and real-time performance support for automated operations. We view this as an important extension of our human performance work primarily because we have observed that military users of automated systems tend to conduct training for these new systems in much the same manner that they did for earlier manual systems. By doing do, trainees do not learn how to take advantage of the performance enhancing effects of automation. Also, older methods for providing operators with on-line performance support (e.g., extensive tabs and pull-down menus) will likely not prove effective in a real-time command and control setting. Operators will not have time to activate displays and browse through them. Clearly, new modes for providing on-line performance support are required. Our research objectives in the training and performance support area are discussed in the paragraphs to follow.

Training and Aptitude Requirements. In the previous section, we remarked that one of our research objectives involves a new look at operator performance requirements in automated systems. Based on previous experience, we are aware of the difficulty in determining training requirements for automated systems, particularly early-on during the system development process before system prototypes are available. Standard front-end analysis methods applied to automated systems will result in a task inventory. On the surface, many of the resulting tasks do not appear different from the operator tasks found in earlier manual systems. The differences between human performance requirements in automated versus manual processing are only apparent when a detailed task analysis is performed. Many operator tasks in an automated environment are, however, highly cognitive in nature. Analyzing such tasks using current methods for cognitive task analysis can be a time-consuming and expensive undertaking.

Recently, we have explored the notion of a Generic Activity Model, or GAM within the context of a Training Impact Analysis (TIA) [i.e., one of the analyses subsumed under the US Army's Training Effectiveness Analysis program] for the National Missile Defense system and have met with encouraging results (Hawley, Frederickson, and Baker, 1993). A GAM is a generic training analysis model, or template, for use with tasks of a given category. We noted above that Sheridan has identified five residual operator functions that, to some extent or another, are always present in an automated person-machine setting. One of our research directions is to explore the notion of a GAM for each of Sheridan's residual functions. The GAMs could provide a framework for rapidly identifying training and aptitude requirements for automated performance environments. In the present context, the term aptitude refers to the skill and knowledge prerequisites

that operator trainees must bring with them to the training setting.

Training Strategies for Automation. Automated systems are complex sociotechnical systems that involve both human and machine components. Human performance requirements in automated processing often are complex, even at the lowest system nodes. In manual systems, operators are able to develop their skills progressively as they move from a simple operating environment to more complex ones. Automated systems, on the other hand, do not require and sometimes prevent active participation by the operators in the control process. Given this situation, the question then becomes, "How will operators acquire and maintain a suitable skill base in such a task performance environment?" Research and experience indicate that the development of complex cognitive skills such as those required in an automated performance environment must be developed progressively, as in previous manual systems. Moreover, it appears necessary to move through each stage in the progression from novice to expert. The skill progression process can be made more efficient but it is necessary to go through all of the steps. Bainbridge (1987) remarks that in the present generation of automated systems, we may be "riding on the skills of former manual operators." She cautions that future generations of operators cannot be expected to have these manually-developed skills. Hopkin (1992), in a reference to automated air traffic control, also notes that there is a "large cognitive difference" between a controller who develops a solution personally and one who chooses a solution from a set of computer-generated alternatives. Choosing a solution from a set of computer-generated alternatives does not require the depth of understanding required to formulate a solution personally.

The evidence cited in the previous paragraph suggests that a new look at training strategies for automated operations is in order. Issues of interest here include: (1) the role of training in manual processes within an overall program concerned with training for automated operations, (2) ways to increase the efficiency of the progression from novice to expert in automated systems, and (3) requirements for skill maintenance training.

There is an emerging view that the real value of expert systems technology lies in *allowing relatively unskilled people to operate at nearly the level of trained experts* (Hammer and Champy, 1993). A dynamic task off-loading aid coupled with an intelligent JPA might result in satisfactory levels of person-machine system performance while significantly reducing the training time and resources currently required to produce a journeyman-level command and control operator. Cost and resource savings could be obtained by identifying the human skills and knowledge actually required for system control at a given node — assuming an effective task off-loading aid and an intelligent JPA — and then training operators in system operations *using these aids*. Air defense decision makers would, however, have to recognize that more complex control decisions must become the responsibility of higher-level command nodes where more skilled controllers are located.

The real issue here is, "How much is the Army willing to pay for skill

redundancy at the lower levels of the command and control hierarchy?" There is no clear answer to this question. Rather, the issue must be approached as a complex trade-off between the operational benefits of control redundancy versus the training costs associated with providing that redundancy along with an assessment of the likelihood that human interventions will be effective when they are required.

Real-Time Performance Support. The widespread use of knowledge-based processing in future systems will create many possibilities for intelligent, embedded performance support. Rasmussen (1986), Sheridan (1992), and others discuss at length the nature of the performance support that must be provided to supervisory controllers. The primary issue that remains unresolved is how to provide operators with performance support for KBB and higher-order RBB within the time lines dictated by the unfolding tactical situation. Given the time lines involved in air defense command and control operations, operators will not have time to browse through a large number of help options. They will require performance support "at their fingertips," so to speak.

The current generation of computer-based JPAs do not operate in anywhere near the time frames required for support of air defense command and control. In a sense, JPAs for future air defense command and control systems must perform in the manner of a personal decision-making assistant. To meet real-time performance demands, these JPAs must almost anticipate operator information and decision-making requirements, as if both operator and machine are following a common script. Some of the recent work in common mental models (e.g., Rouse, Cannon-Bowers, and Salas, 1992) might point the way to the development of JPAs suitable for real-time command and control applications.

5.0 DISCUSSION

One of the ironies of automation is that increasing levels of machine processing often result in increased difficulty for human operators. We noted earlier that human performance problems are often attributed to (1) loss of situational awareness and (2) skill decay. In many instances, the possibility for short-term loss of situational awareness and longer-term skill decay as a consequence of automated processing result from an unreasonable operator task set. An unreasonable task set can arise because of inappropriate levels of workload (too little for the operator to do in the case of automation) or an incoherent residual task set. Problems with operator workload and task set coherence typically accompany a design approach termed, "Let the machine do it." That is, let us automate everything that the technical state-of-the-art will permit and that we can afford. Such an approach often results in human operators being left with whatever cannot be automated. The result is a fragmented, difficult-to-perform job for which training is also a problem.

The phenomena described in the previous paragraph have been known for some time. Early system designers sometimes attempted to circumvent the problem by requiring operators to make periodic log book entries. Their logic was that if

operators are required make a periodic assessment of system status, then they will be required to maintain or at least periodically re-establish situational awareness. However, as soon became apparent, operators can make log entries without fully attending to their significance. Some observers such as Wesson (1981) have argued that the problems associated with loss of situational awareness and skill decay are serious enough to consider stopping short of the level of automation that is technically possible in a given situation. If operators are to function effectively as supervisory controllers, they must have something meaningful to do during routine operations. However, leaving the operators with something consequential to do might require artificially limiting the level of automation employed in order to keep operators meaningfully in the control loop.

The operators' role in Patriot semi-automatic processing is a good example of an attempt to implement this latter strategy. In semi-automatic mode, Patriot operators are kept in the control loop through a requirement to manually perform a number of keyboard entries and switch actions that would better be left to the machine. In our view, there is not much difference between making rote log entries and mechanically making keyboard entries and switch actions in response to machine decisions. Decision makers are left with the impression that human operators have a meaningful role in the control process. One could question, however, whether that effectively is the case. Positive control implies considerably more than perfunctory participation by operators in the control process.

Automation theorists have recognized for some time that a potential solution to the twin problems of loss of situational awareness and skill decay is flexible automation. A flexible automation scheme is one in which both the level and style of automation are variable as a function of operating conditions. Operators can choose the level of automation suitable to the task at hand (e.g., near-manual through fully automatic) and can also vary automation style across several dimensions. In theory, flexible automation will reduce the potential for loss of situational awareness if operators adjust the level of machine aiding to bring about a requisite level of involvement in the control process. However, we do not know what an appropriate level of involvement is and there is no evidence that operators will perform so rationally. Similarly, flexible automation will, in theory, prevent skill decay because even in a highly automated setting operators will periodically be required, or will choose, to perform in a manual role. Manual processing requirements will reinforce manual processing skills, or so the story goes. Again, we have no empirical evidence to support this contention, nor do we know how much manual processing is required to maintain skill proficiency. A skeptic would say that in both situations we might be making unreasonable assumptions about the rationality of operator behavior.

Although our experience with flexible automation is limited, several recent studies suggest that flexible regimens are not a panacea for the human performance problems associated with automation. Wood (1993) remarks, for example, that the potential for automation mode (e.g., level and style) changes may actually have an adverse performance impact on human operators. He notes that it is easy for the

operator to lose track of what mode the automated system is in. Hence, rather than having less workload, the operators may actually have more since they must (1) track the automation mode and (2) know about each mode and option. Further, in a simulated air defense command and control task, Adelman, Cohen, Bresnick, Chinnis, and Laskey (1993) report that the use of an intelligent interface (i.e., an embedded expert system used somewhat like that proposed for APAWS) allowed operators to focus their attention where the rule base indicated it was required most. Operators were able to expand the set of tracks handled by the machine, thus giving them more time to select and examine high priority situations. At the same time, however, their handling of less critical, yet important situations was inferior to the unaided case. Both of these studies suggest that flexible automation has both an up-side and a down-side and that blind application of flexible automation in real-time process control could lead to some unpleasant surprises.

In spite of the problems noted in the previous paragraphs, flexible automation holds enough promise to warrant further study. Also, recent developments in software technology have rapidly pushed flexible automation from theory to reality. A number of approaches to flexible automation are possible, but there is little theoretical or empirical guidance concerning which approach is best for what applications. Some observers such as Moray (1990) argue that it may prove impossible to develop general guidelines for flexible automation (or dynamic function allocation) that are applicable in all situations. If this turns out to be the case, then it may prove necessary to develop domain specific guidelines empirically — by trial and error. APAWS will be a useful tool for the development and evaluation of potential concepts for human-automation integration in future HIMAD command and control systems.

In the APAWS effort, we have taken Sheridan (1992) literally in that the core conceptual issue in supervisory control is the effective partitioning of control intelligence between human and machine components. We are attempting to formalize this notion in the form of the IRBSC concept. IRBSC is based on the premise that the proper role for human supervisory controllers is to retain higher level tasks and determine the machine's goals while the majority of moment-to-moment operations are handled by the machine. Under the IRBSC concept, if the operators decide to participate in moment-to-moment operations, they explicitly have to take back selected activities from the machine. The operator has the flexibility to set the level and style of automation anywhere that personal preferences or the operating situation dictate. Whether this capability will make any difference in terms of system effectiveness or training effectiveness and efficiency remains an empirical issue, however.

Air defense decision makers are faced with a significant challenge. A steady progression of highly-automated, real-time command and control systems is coming on line. Our observation is, however, that system developers generally have failed to recognize and accommodate the human performance and training implications of automated operations. Concepts of operation for humans in control of automated, real-time command and control systems have not kept pace with machine

technology. The unfortunate fact is that all too often developers continue to conceive of the operators of such systems simply as manual controllers who now are in charge of more complex equipment. Current training methods, for the most part, also reflect this orientation. As a consequence, operators often do not know how to use automated systems to their full potential and the system's potential is not realized. Advanced command and control systems employing various forms of flexible automation will soon be standard fare within the HIMAD arena. One objective of the APAWS effort is to increase the likelihood that these systems are designed in accord with principles of effective human-automation cooperation.

As a final comment on automation and contemporary command and control, consider Swamidass' (1993, p. 69) remark that the "proliferation of microprocessors has stood the meaning of automation on its head." He notes that the term "automation" once referred to inflexible technology employed in high-volume, low-variety, and low-cost processing. Today, thanks to the proliferation of inexpensive information technology, automation means increased flexibility. Swamidass also remarks, however, that the flexibility inherent in this new technology is worthless unless it can be exploited by the organization — the soldiers, doctrine, tactics, and leadership — surrounding it. Otherwise, the resulting capability is "dumb." In dumb automation, a flexible, advanced technology replaces an inflexible one, but inherits all the people, procedures, and organization used in managing the older technology. Swamidass notes that countless case studies have shown the ineffectiveness of dumb automation. Modern automation technology will pay off when it is coupled with a flexible environment, well-trained soldiers, and effective leadership. Smarter soldiers and good leadership are no substitute for advanced technology. But neither can technology substitute for good soldiers, effective leadership, and organizational flexibility.

REFERENCES

Adelman, L., Cohen, M. S., Bresnick, T. A., Chinnis, J. O., & Laskey, K. B. (1993, June). Real-time expert system interfaces, cognitive processes, and task performance: An empirical assessment. Human Factors, 35(2), 243-261.

Bainbridge, L. (1987). Ironies of automation. In J. Rasmussen, K. Duncan, and J. Leplat (Eds.), New technology and human error. New York: Wiley.

Brett, B. E., & Allender, L. (1990). Performance assessment capability feasibility study (ARI Research Note 90-127). Alexandria, VA: U.S. Army Research Institute for the Behavioral and Social Sciences. (AD A227 290)

Department of the Army. (1993, June). FM 100-5, Operations. Washington, DC.

Gersh, J. R., & Hamill, B. W. (1991). Cognitive engineering of rule-based supervisory control systems: Observations concerning users' task performance. Proceedings of the 1991 IEEE International Conference on Systems, Man, and Cybernetics. Charlottesville, VA: IEEE Systems, Man, and Cybernetics Society.

Hamill, B. W., & Gersh, J. R. (1991). Understanding rule-based supervisory control. Proceedings of the 1991 Joint Directors of Laboratories Basic Research Group Symposium on Command and Control Research. Washington, DC: National Defense University.

Hamill, B. W., & Gersh, J. R. (1992). Decision-making performance in rule-based supervisory control: Empirical development of a cognitive process model. Proceedings of the 1992 Joint Directors of Laboratories Basic Research Group Symposium on Command and Control Research. Washington, DC: National Defense University.

Hammer, M., & Champy, J. (1993). Reengineering the corporation. New York: HarperBusiness.

Hawley, J. K., Frederickson, E. W., & Baker, P. J. (1993). National missile defense system training impact analysis. Fort Bliss, TX: U.S. Army Air Defense Artillery School.

Hawley, J. K., Howard, C. M., & Martellaro, A. J. (1982). Optimizing operator performance on advanced training simulators: Preliminary development of a performance assessment and modeling capability (ARI Technical Report 573). Alexandria, VA: U.S. Army Research Institute for the Behavioral and Social Sciences. (AD A133 135)

Hawley, J. K., Strub, M. E., & Lockhart, J. M. (1993). Automation and supervisory control: An air defense command and control research and training perspective. Proceedings of the 1993 Joint Directors of Laboratories Basic Research Group Symposium on Command and Control Research. Washington, DC: National Defense University.

Hopkin, V. D. (1992, June). Human factors issues in air traffic control. Human Factors Society Bulletin, 35(6), 1-4.

Howell, W. C. (1993). Engineering psychology in a changing world. Annual Review of Psychology, 44, 231-263.

Kirlik, A. (1993, June). Modeling strategic behavior in human-automation interaction: Why an "aid" can (and should) go unused. Human Factors, 35(2), 221-242.

Moray, N. (1990). Mental models of complex systems. In N. Moray, W. R. Ferrell, and W. B. Rouse (Eds.), Robotics, control, and society. New York: Taylor & Francis.

Obermayer, R. W. (1991). Observing system for advice, critique, and review (OSACR) (Technical Report). Williams AFB, AZ: U.S. Air Force Armstrong Laboratory.

Rasmussen, J. (1986). Information processing and human-machine interaction: An approach to cognitive engineering. New York: North Holland.

Rouse, W. B., Cannon-Bowers, J. A., & Salas, E. (1992). The role of mental models in team performance in complex systems. IEEE Transactions on Systems, Man, and Cybernetics, 22, 1296-1308.

Shanit, J., Chang, T.-C., & Salvendy, G. (1987). Technical and human aspects of computer-aided manufacturing. In G. Salvendy (Ed.), Handbook of human factors. New York: Wiley.

Sheridan, T. B. (1992). Telerobotics, automation, and human supervisory control. Cambridge, MA: MIT Press.

Swamidass, P. M. (1993, September). Manufacturing a la carte—agile assembly lines, faster developmental cycles: Technology, people, and management (Special issue on manufacturing technology). IEEE Spectrum, 30(9), 68-69.

U.S. Army Air Defense Artillery School. (1991). Air defense tactical operations center (ADTOC) user functional description (UFD). Fort Bliss, TX.

U.S. Army Air Defense Artillery School. (1992). Air defense artillery command, control, communications, and intelligence (C3I) master plan. Fort Bliss, TX.

Wesson, R. B. (1981). An alternative scenario for air traffic control: Shared control (Technical Report P-6703). Santa Monica, CA: The Rand Corporation.

Wood, D. D. (1993, June). Clumsy automation, practitioner tailoring, and system disaster. CSERIAC Gateway, 4(2).